

What is claimed is:

1. A method for storing a computer program (21) in a program memory (15, 20) of a control unit (10), the computer program (21) being stored according to predefinable rules in specific memory areas of the program memory (15, 20), wherein predefinable information (22), which is used to transfer the control unit (10) into a defined state, is stored in unused memory areas of the program memory (15, 20) where the computer program (21) is not stored.
2. The method as recited in Claim 1, wherein the control unit (10) is reset by executing the predefinable information (22) on a computing unit (11), in particular on a microprocessor, of the control unit (10).
3. The method as recited in Claim 1 or 2, wherein an interrupt service routine is invoked by executing the predefinable information (22) on a computing unit (11), in particular on a microprocessor, of the control unit (10).
4. The method as recited in one of Claims 1 through 3, wherein an error handling routine is invoked by executing the predefinable information (22) on a computing unit (11), in particular on a microprocessor, of the control unit (10).
5. The method as recited in Claim 3 or 4, wherein the control unit (10) is reset at the end of the routine.
6. The method as recited in one of Claims 1 through 5, wherein the predefinable information (22) is stored in all unused memory areas of the program memory (15, 20).
7. The method as recited in one of Claims 1 through 6,

wherein at least one unused memory area of the program memory (15, 20) is completely filled using the predefinable information (22).

8. The method as recited in one of Claims 1 through 6, wherein the predefinable information (22) is stored at predefinable intervals in at least one unused memory area of the program memory (15, 20), the part of the unused memory area in which the predefinable information (22) is not stored not causing jumps or endless loops.

9. The method as recited in Claim 8, wherein the predefinable information (22) is stored at regular intervals in the at least one unused memory area of the program memory (15, 20).

10. The method as recited in one of Claims 1 through 6, wherein the predefinable information (22) is stored at the end of at least one unused memory area of the program memory (15, 20), the part of the unused memory area in which the predefinable information (22) is not stored not causing jumps or endless loops.

11. A device (25) for storing a computer program (21) in a program memory (15, 20) of a control unit (10), the device (25) having first means for storing the computer program (21) in specific memory areas of the program memory (15, 20) according to predefinable rules, wherein the device (25) has second means for storing predefinable information (22), which transfer the control unit (10) into a defined state, in unused memory areas of the program memory (15, 20), in which the first means have not stored the computer program (21).

12. The device (25) as recited in Claim 11,

wherein the second means are implemented as a hexadecimal editor.

13. The device (25) as recited in Claim 11 or 12, wherein the device (25) has means for executing a method as recited in one of Claims 2 through 10.

14. A control unit (10) having a computing unit (11), in particular a microprocessor, and a program memory (15, 20), on which a computer program (21) is stored in specific memory areas of the program memory (15, 20) according to predefinable rules, wherein, in unused memory areas of the program memory (15, 20), where the computer program (21) is not stored, predefinable information (22) is stored, which is used to transfer the control unit (10) into a defined state.

15. The control unit (10) as recited in Claim 14, wherein the predefinable information (22) is stored in the unused memory areas of the program memory (15, 20) according to a method as recited in one of Claims 2 through 10.